LabState version 1.0
A Graphic State Notation Program for Animal Behavioral Study

User's Guide to Experimental Method Editor

AniLab Software & Instruments Co., Ltd.
Copyright 2006-2014
1st Edition, Sept. 10, 2014
On-line: www.anlab.cn
E-mail: sales@anilab.cn

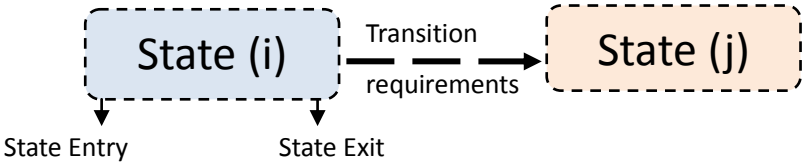
TABLE of CONENTS

1. Introduction 1
2. Creating a simple FR1 Method 4
3. Session 8
4. Components 10
5. States 11
    5.1 State Entry 12
        5.1.1 Output Device 12
        5.1.2 Sound 13
        5.1.3 Random Output 14
        5.1.4 Touchscreen 15
    5.2 Set Session Variables 16
        5.2.1 State Entry: Counter& Timer Variables 16
        5.2.2 State Entry: Temporary Counter 16
        5.2.3 State Entry: Analog Variables 17
        5.2.4 State Entry: Save Temporary Variables 17
        5.2.5. State Entry: Video Tracking Variables 17
        5.2.6. Digital ON-Event: Counter &Timer Variables 18
        5.2.7. Digital OFF-Event: Counter &Timer Variables 18
        5.2.8. Analog High/Low Event: Counter &Timer Variables 18
        5.2.9. Video Tracking Zone-Event: Counter &Timer Variables 19
        5.2.10. Touch-Screen ZONE: Counter Variables 19
        5.2.11. Touch-Screen IMAGE: Counter Variables 19
    5.3 State Transition 20
        5.3.1 Single digital-event: input-ON 20
        5.3.2 Digital Event: input-off 21
        5.3.3 Digital-Event: Input-ON time 22
        5.3.4 State Timer 22
        5.3.5 State Counter 22
        5.3.6 Next State List 23
        5.3.7 Counter Variable 23
        5.3.8 Timer Variable 23
        5.3.9 Computed Variable 23
        5.3.10 Video Tracking Zone 24
        5.3.11 Multiple Digital Event: Input ON/Off 24
        5.3.12 Exclusive Digital Event: Input ON 24
        5.3.13 Analog Event: high/low 24
        5.3.14 Keyboard Event 25
        5.3.15 Touchscreen event 25
    5.4 State Exit 26
6. Sample Methods 27
    6.1 Second order schedule 27
    6.2 Five-choice serial reaction time task 27
    6.3 GO/NO-GO 27
    6.4 Pre-pulse Inhibition of acoustic startle reflex (PPI) 27
    6.5 Active-avoidance: shuttle box 27
    6.7 Touch-screen: 5-CSRT 27

1. Introduction

LabState is a windows program designed for experimenters to create interactive protocols for behavioral experiments using state logic. It is a “point and click” windows program using a graphic screen that presents options to be selected by the user. You need not to learn a language to write scripts. It is all there in each window.

In LabState, a series of states is called component, a session (or experiment) may contain several components. In each component, the experiment moves from one state to another. Each **state** specifies a set of device controls, and a set of transition requirements which cause the program to exit the current state and move on to the next state.



STATE ENTRY

Upon entry of a state, a set of controls could be configured:

- Digital output: light, tone, infusion pump, etc.
- Analog input: start/stop signal recording
- Analog output: optogenetc LED current, etc.
- Sound mixer: play pure sine tone (20-20Kh) , white noise, or wav file, in 8 channels
- Session variables:
  - Counter variable: increase or decrease
  - Timer variable: start a timer or read/stop a timer
  - Analog variable: capture averaged analog input signal.
  - Video tracking variable: start/stop video tracking
- Screen control: draw picture or text in the screen

STATE EXIT

- Stop session
- Stop the current component or start next component(s).

TRANSITION REQUIREMENTS

When a transition requirement is meet, the program moves on to next state. The state transition requirements are:

- Time
- State entry count
- Digital input: ON/OFF/Keep-ON time
- Analog input: high/low above the baseline
- Video tracking: enter/exit a zone, moving/resting
- Key/mouse input
- Touch-screen input
- Counter Variable

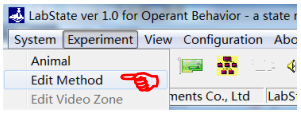
About State notation:
State notation has been increasingly utilized to describe and implement behavioral procedures since its adaptation for this purpose in 1970. The original version describes states as unique segments of an experimental procedure, accompanied by specified stimulus conditions. Transitions among states are triggered by inputs from the subject, or by the passage of time, and may be the occasion for changing stimulus conditions, recording data, or per-forming other operations. Extensive usage has suggested a number of possible improvements, and the notation therefore has been expanded and modified. Revisions recently incorporated in state notation increase its power as a descriptive device for effectively communicating the procedural details of reinforcement contingencies.
Arthur G. Snapper, Ronald M. Kadden, Geoffrey B. Inglis. State notation of behavioral procedures. Behavior Research Methods & Instrumentation. May 1982, Volume 14, Issue 3, pp 329-342

Experimental Method Editor Window

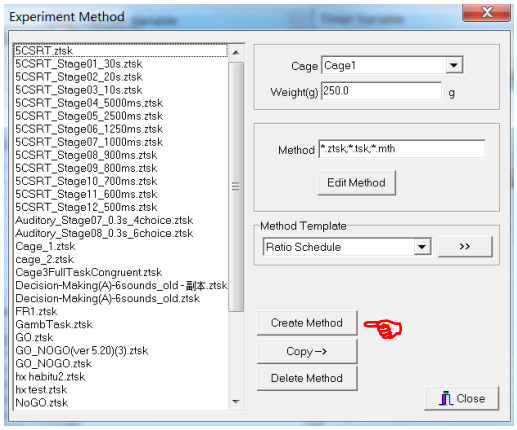
In the main window of LabState, click the shortcut key "*Edit Method*",

LabState ver 1.0 for Operant Behavior - a state notation program for animal behavi
System Experiment View Configuration About

or the menu "Experiment - Edit Method",



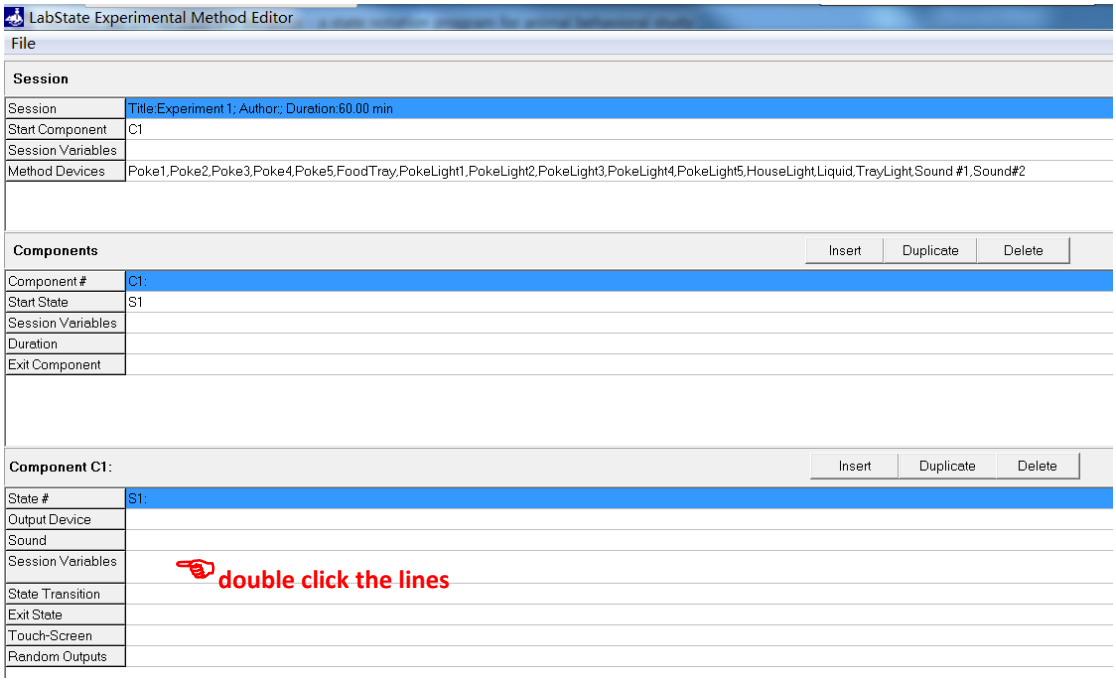
the "Experiment Method" window appears.



In this window, you may chose:

- create a new method
- edit an existing method
- create a method by using the method template
- copy a method
- delete a method

Click the "Create Method" button, the "Experimental Method Editor appears.



You may ***"double click"*** any line in the tables to enter a sub-window for the method options. There are three panels (tables) in the method editor window: Session, Component(s), State(s).

Each session (or an experiment) has at least one component (maximum 16), and each component has at least one state (maximum 24). You can add/delete/duplicate component or state by clicking the corresponding button.

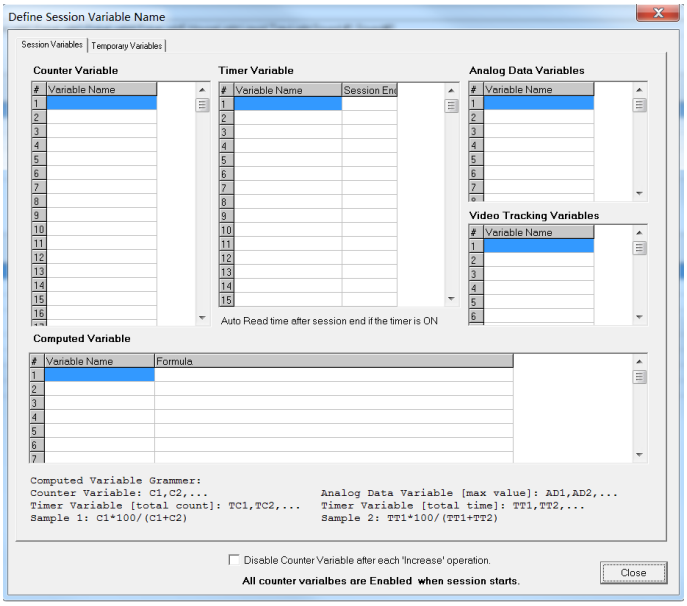
When a session starts, the components can be active concurrently or in order. In each active component, only one state is active at one time.

Session Variables:

Generally, in a simple experiment, you need not to define session variables (or user defined variables), LabState will collect any events (on/off) such as digital inputs/outputs, sound devices, component/state, to be saved in the data file.

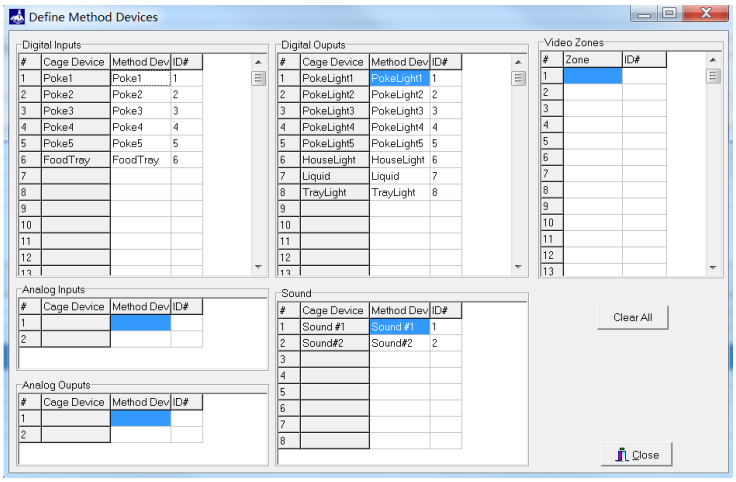
You may define you own variables in the following types:

- Counter variables: correct response, incorrect response, response omission, etc.
- Timer variables: response latency, etc.
- Analog Data Variables: used for acoustic startle reflex experiment
- Video Tracking Variables
- Computed Variables



Method Devices:

When you create a new method, the devices in the system will automatically be assigned to the method editor, including digital inputs/outputs, analog inputs/outputs, sound devices, video tracking zones.



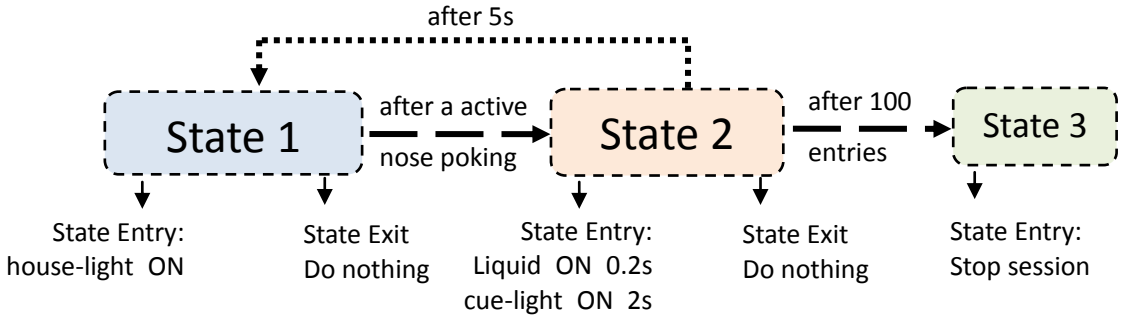
If you want the method file can be used in other experimental system which have different hardware configuration, you can redefine the method device with a name easy to understand, such as: active response, cue-light, reward, etc.

When a method is running, the system devices are controlled according the method device ID#, irrelative of the device name. The method device ID# corresponding to the system device ID#.

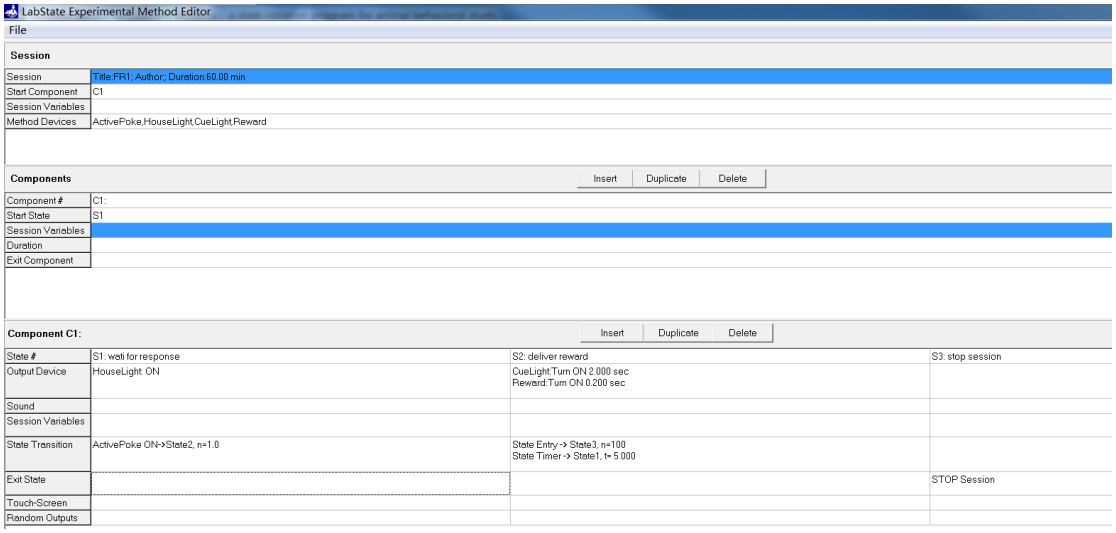
2. Creating a simple FR1 Method

For a fixed ratio 1 (FR1) reinforcement schedule, the state-notation logic could be showed in the following diagram. A session starts from state 1, a house-light turn on, after the animal make a nose-poking response in the active nose poke, the program goes to state 2. In state 2, the liquid dispenser and the cue-light (above the active lever) are on. After a period of time (5 seconds), the program goes back to state 1 and the procedure repeats. The house-light remains on for the entire session because the house-light does not turned off ever since it is turned on in state 1. After the animal earned 100 liquid deliveries (100 entries of state 2), the program goes to state 3 in which the program stops and the experimental data are saved.

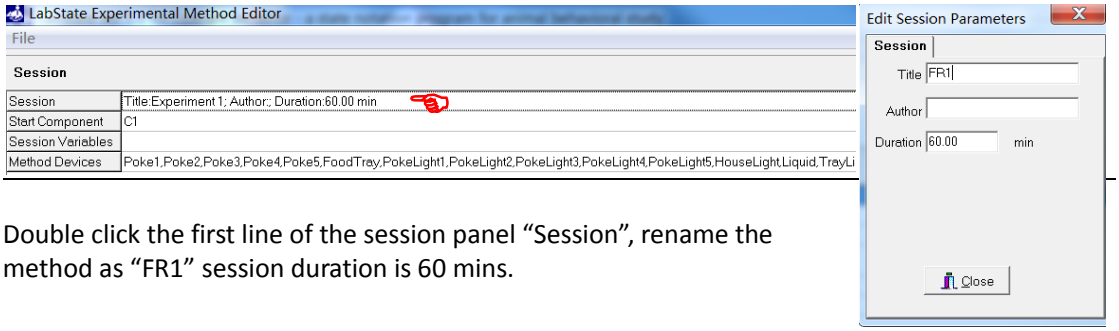
FR1 schedule:



The Method Editor window: FR1



<u>Step 1:</u> *define session duration*



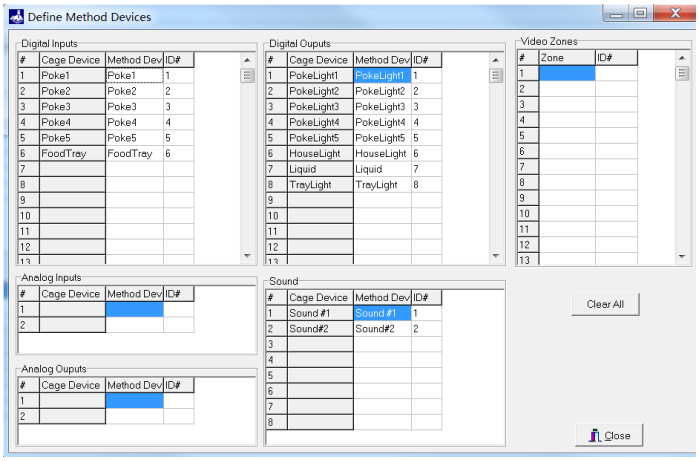
Double click the first line of the session panel “Session”, rename the method as “FR1” session duration is 60 mins.

Because only one component is needed for a FR1 method, there is no need to edit the second line of the session panel "Start component". The only component 1 is automatically active when a session starts.

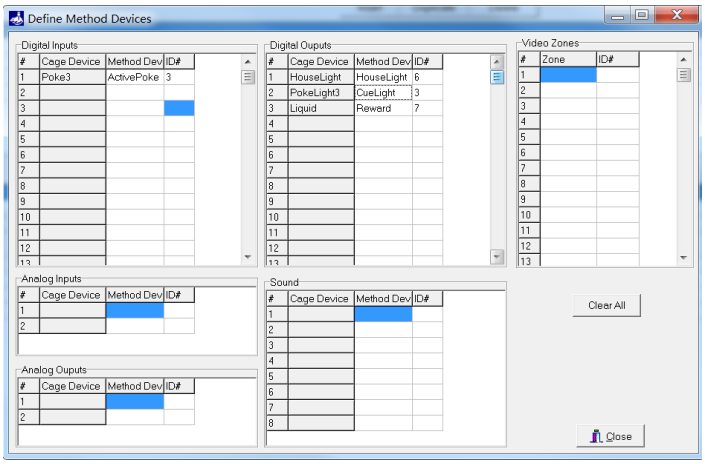
The third line of the session pane is "Define session variables", there is no need to define a session variable for a FR1 method.

Step 2: define method devices

You may choose to skip this, or you can rename the method devices.

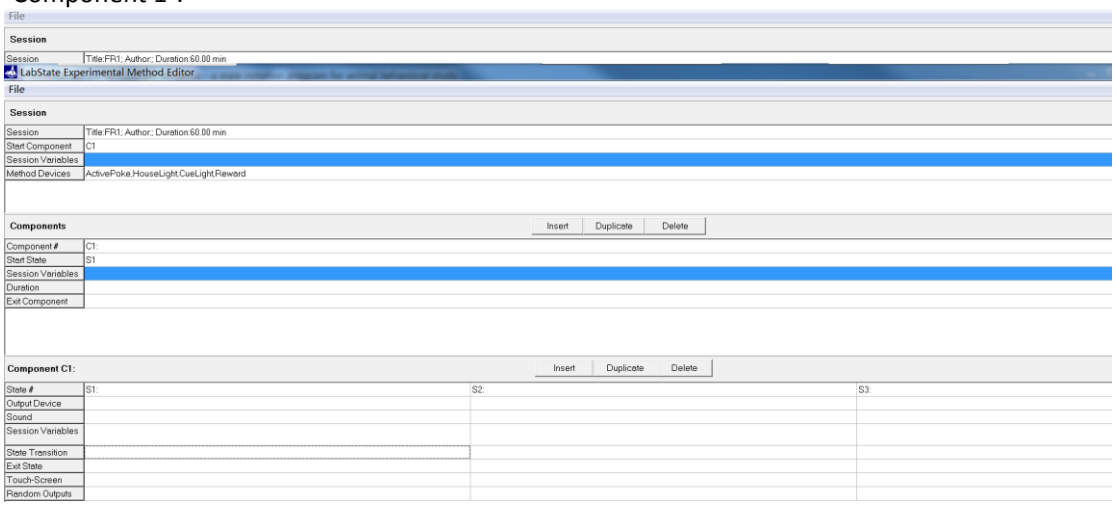


The method device can be rename as following for easy understanding. The center nose poke (Poke3) is defined as "active poke", and the light in Poke3 is defined as "cue-light". The liquid dispenser is defined as "reward".

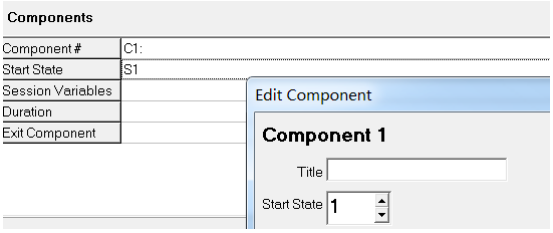


Step 3: add new states

Click the "insert" button twice in the state pane, we now have three states (S1 - S3) for "Component 1".



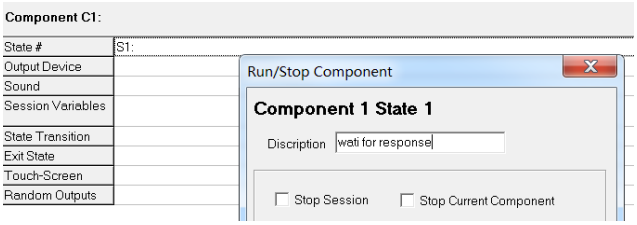
You may choose which state is active first when “component 1” start.
Double click the first or second line of the components panel.
State 1 is set as the start state.



Step 4: *name the states*

Double click the state panel: first line and S1 - S3 column.

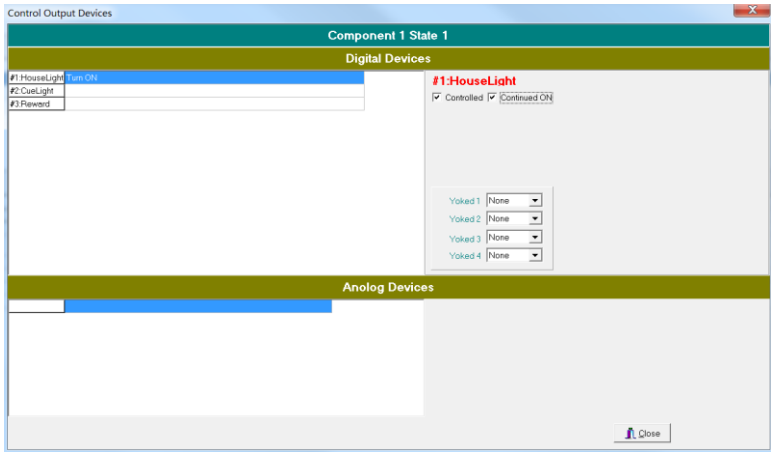
State1 is named as “wait for response”, State 2 as “deliver reward”, State 3 as “stop session”.



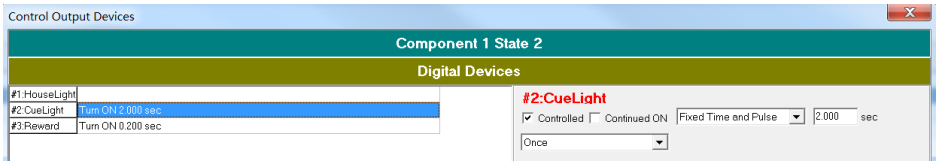
Step 5: *Editing output devices*

Double click the state panel: second line (output device) and S1 - S3 column.

S1: turn on the houselight

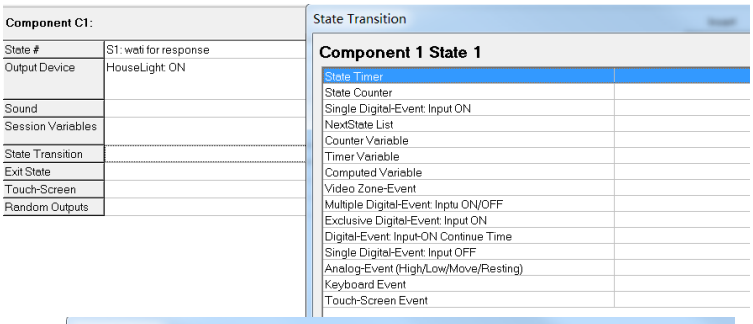


S2: turn on “cue-light” 2s, “reward” 0.2s

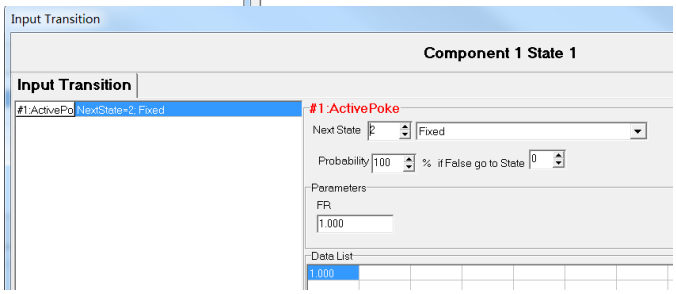


Step 6: *Editing state transition requirements*

Double click the fifth line (state transition) of state panel



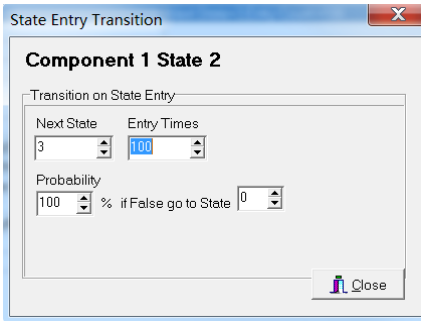
For S1, in the state transition window, double click the third line (single digital-event input ON). “ActivePoke” is set as “next state = 2”,and “FR = 1”.



For S2, in the state transition window,

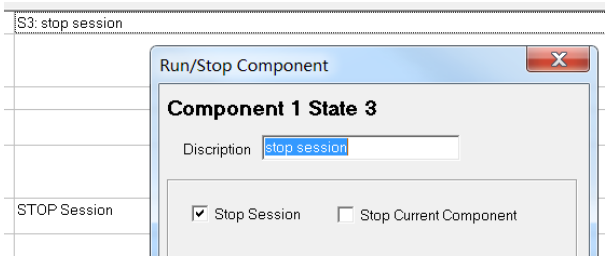
double click the first line (State Timer). Next state is 1 an FT = 5s

double click the second line (State Counter). Next state is 3 and entry times = 100.

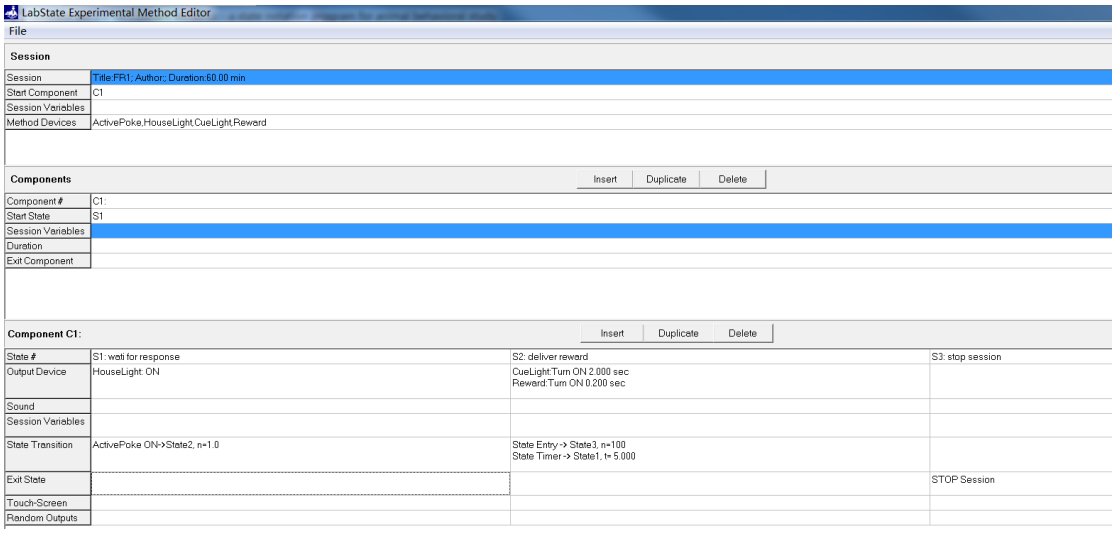


<u>Step 7:</u> Editing state 3: stop session

double click the first line of the state panel for S3. Check the “stop session” box.



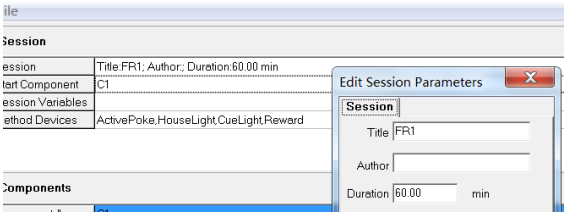
The finished FR1 method editor windows is as following:



3. Session

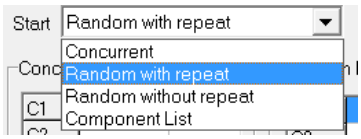
In LabState, the top level in an experimental method is session. A session is composed of component(s) and the component is composed of state(s).

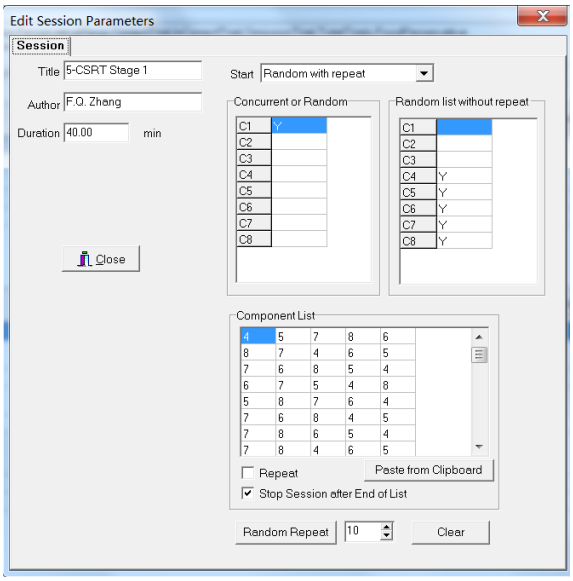
"Session duration" sets the running time of a session. The session will automatically stop after the "duration". A session can also be stopped by give a "stop session" command in a state.



If a session only has one component, the component will be active when the session starts. The "Start Component" will be component 1 (C1).

If a session has more than one component. You can set the running status of each component when the session starts.





There are four ways to set the running status for the components:

- Concurrent: the components can be active concurrently when a session starts. The "concurrent or random" list specifies the component is active when set as "Y".
- Random with repeat: randomly select one component from the "concurrent or random" list.
- Random without repeat: randomly select one component from the "Random list without repeat".
- Component list: select the first component from the "Component List".

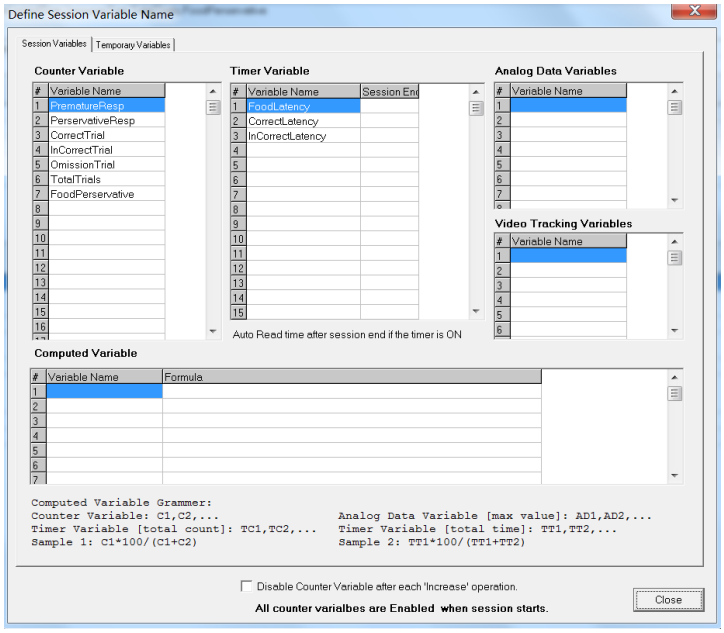
"**Session Variables**" are custom defined variables to collect experimental data. You need not to define a session variable for digital input/output ON/OFF events.

You may define you own variables in the following types:

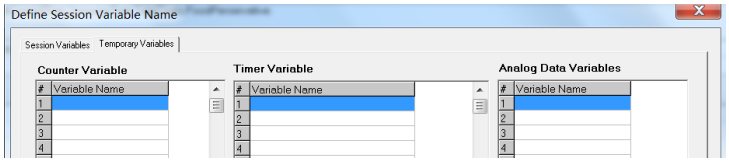
- Counter variables: correct response, incorrect response, response omission, etc.
- Timer variables: response latency, etc.
- Analog Data Variables: used for acoustic startle reflex experiment
- Video Tracking Variables
- Computed Variables

For example, in a typical 5-choice serial reaction time task (5-CSRT), the following session variable are defined:

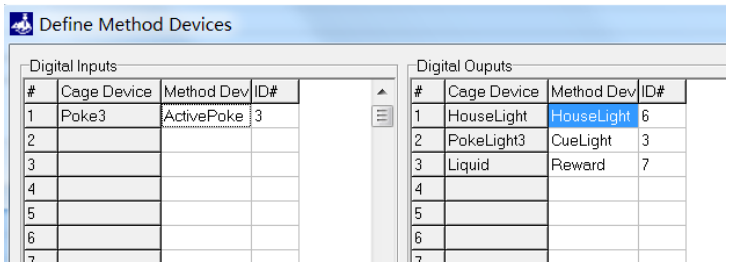
- Counter variables: Total trial, Correct trial, Incorrect trial, Omission trial, Premature response, Perservative response
- Timer variable: Correct latency, Incorrect latency, food intake latency



The session variable also has corresponding temporary variable. Session variables are saved in the data file, but the temporary variables are not. The temporary variables are used to collect temporary data and can be assigned to session variables.



"**Method Devices**" are used to defined devices to be controlled in an experiment.



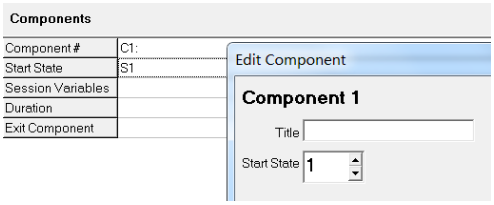
When you create a new method, the devices in the system will automatically be assigned to the method editor, including digital inputs/outputs, analog inputs/outputs, sound devices, video tracking zones.

If you want the method file can be used in other experimental system which have different hardware configuration, you can redefine the method device with a name easy to understand, such as: active response, cue-light, reward, etc.

When a method is running, the system devices are controlled according the method device ID#, irrelative of the device name. The method device ID# corresponding to the system device ID#.

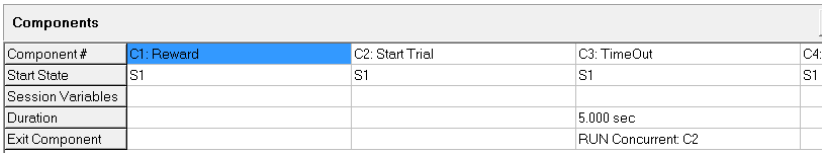
4. Components

You can inset/duplicate/delete a component in the component panel by clicking the corresponding button. The maximum number of components in a session is 16.

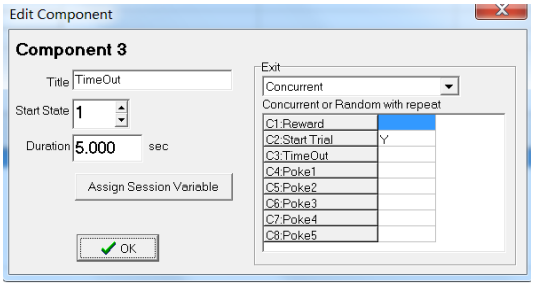


If the session has only one component, there is only one option in the component panel.
"Start State" specified which state is active when the component or session starts.

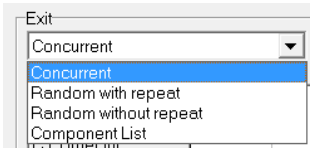
If the session has more than components, you can set the running duration for a component, when the component stops after the duration, you can set the running status of other components.



There are four ways to set the running status for the components:



- Concurrent: the components can be active concurrently when a session starts. The "concurrent or random" list specifies the component is active when set as "Y".
- Random with repeat: randomly select one component from the "concurrent or random" list.
- Random without repeat: randomly select one component from the "Random list without repeat".
- Component list: select the first component from the "Component List".



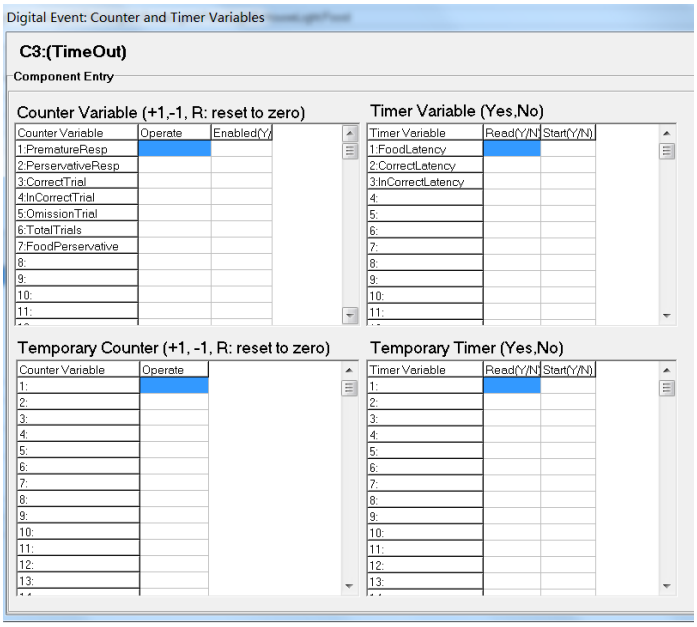
"Assign Session Variable" is used to operate the session variable when the component starts.

Counter Variable has three operations:

- Increase (+1,+2,...)
- Decrease (-1,-2,...)
- Reset to zero (R)

When a session starts, all the counter variable will be enabled automatically and can operated. When it is disabled ("N)", the "+/-" operation is not allowed.

Timer variables can started. When a timer variable is read, it stops automatically.

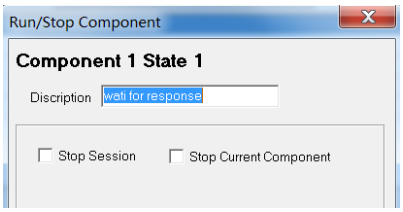


5. States

State is the basic unit of the state logic. In a given time, only one state is active for a component.
You can inset/duplicate/delete a state in the state panel by clicking the corresponding button. The maximum number of states for a component is 24.

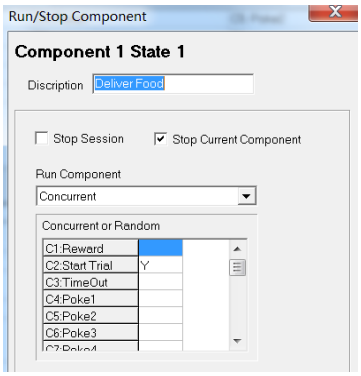
Component C1:	
State #	S1: wati for response
Output Device	HouseLight: ON
Sound	
Session Variables	
State Transition	ActivePoke ON->State2, n=1.0
Exit State	
Touch-Screen	
Random Outputs	

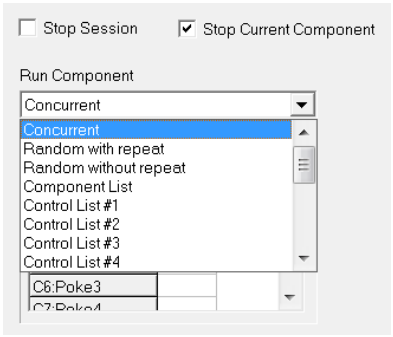
Double click the first line “State #” in the state panel, for a session with only one component, it appears as follow:



The current component as well as the session can be stopped by check the corresponding boxes.

For a session with more than one components, it appears as follow:

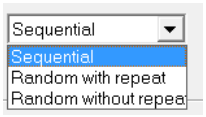




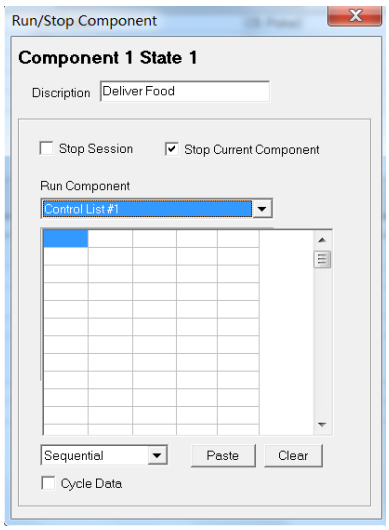
when the “stop current component” is checked, “Run Component” list box will appears.

In addition to the first four operations for the components, you can also used the “Control List”.

The “Control Lists“ are global data list for a session. There are 16 “Control Lists” in a session. Each control list is a set of numbers.



There three way to read a data from the “control list”:
Sequential, random with repeat, and random without repeat.



5.1 State Entry

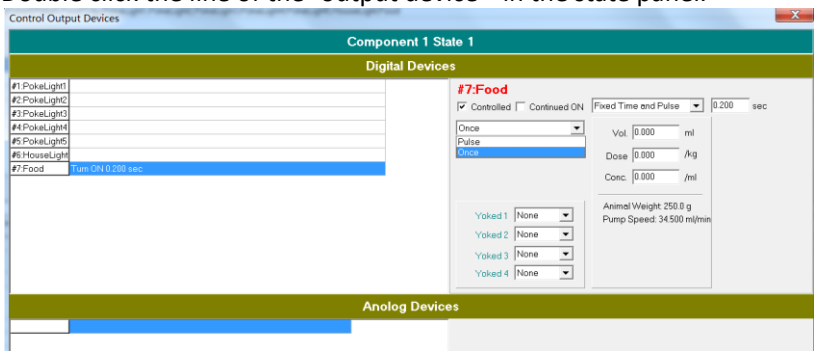
When entering a state, you can control the following devices:

- Output Device
- Sound Device
- Touchscreen
- Random Outputs

and operate the session variables.

5.1.1 Output Device

Double click the line of the ”output device ” in the state panel.

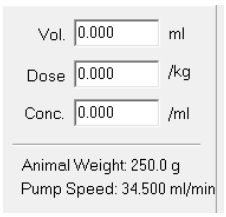


Digital Outputs:

A digital output device is controlled when the “Controlled” box is checked. When it is controlled, it can be turned ON continually by check the “Continued ON”, it will remain ON until turned OFF.

A digital output device can be turned on for a period of time in two ways:
“once”: continue on
“pulse”: on/off alternating

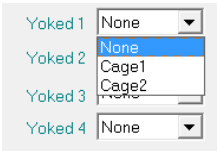
If the output device is an infusion pump (pump speed of the output device is non-zero), you can control it by “vol” or “dose”.



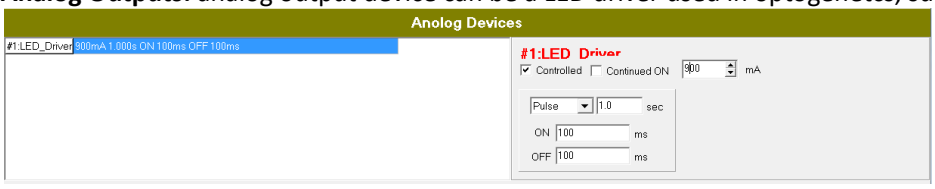
The ON duration (t) is calculated as (seconds):

“Vol”: t = vol [ml] *60 / speed [ml/min]
“Dose”: t = body weight[kg]*(dose[mg/kg] / concentration[mg/ml]) *60/speed (ml/min)

There are four “**Yoking**” controls of digital output devices. When the digital output device is on/off, the same devices (same ID#) in the “yoked” cages (maximum 4 cages) are also turned on/off.

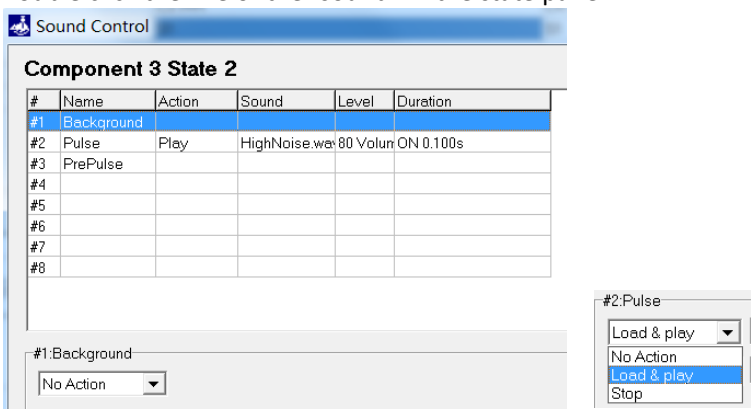


Analog Outputs: analog output device can be a LED driver used in optogenetcs, such as following.



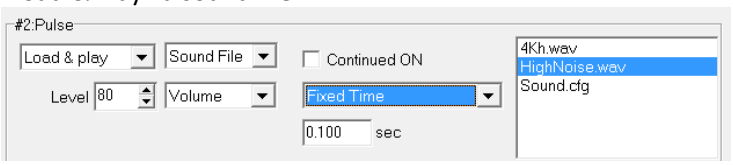
5.1.2 Sound

Double click the line of the ”sound” in the state panel.

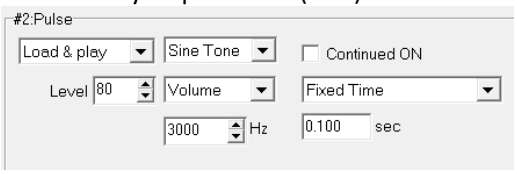


There are sound channel can be played.

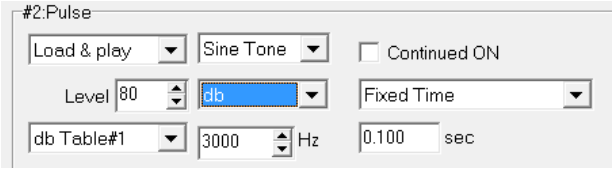
“Load & Play” a sound file:



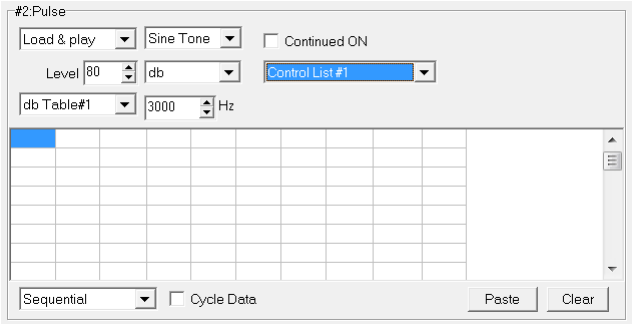
“Load & Play” a pure tone (sine):



The sound level can also be in db as calculated in db Table#

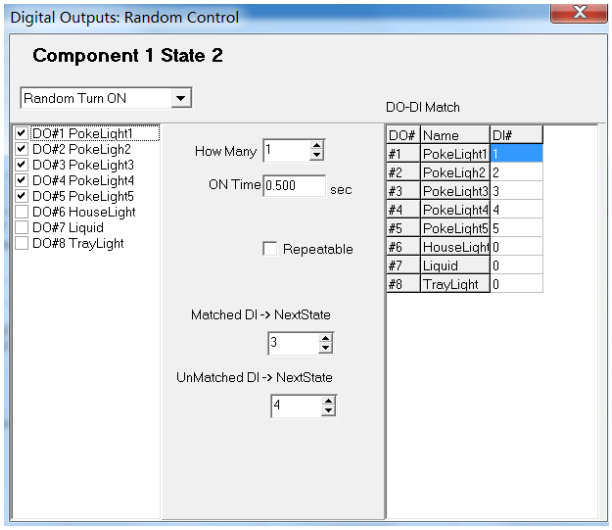


Play duration can also select from a “Control List”.



5.1.3 Random Output

Double click the line of the ”Random Output” in the state panel.
This operation is specifically designed for the digital output control of 5-choice serial reaction time task.



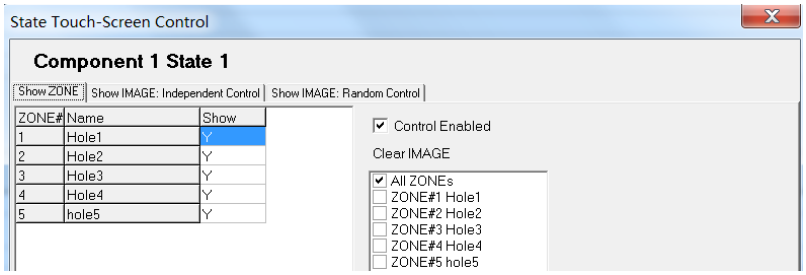
In this example, randomly turn on (0.5s) one device selected from the device list (DO#1-DO#5). State transition is also defined in this window.

The DI-DO match table sets matched digital input (DI) devices for each digital output (DO) device. When an output device is ON, and the matched input device has a response, the current state will exit and moves to the next state of Match DI, otherwise moves to the next state of UnMatched DI.

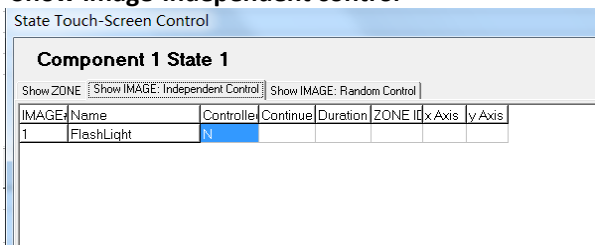
5.1.4 Touchscreen

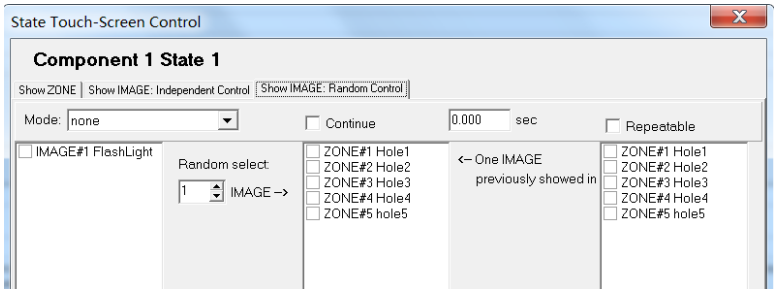
Double click the line of the ”Touch-screen” in the state panel.
There are three panels in the state touchscreen control window.

“Show Zone”

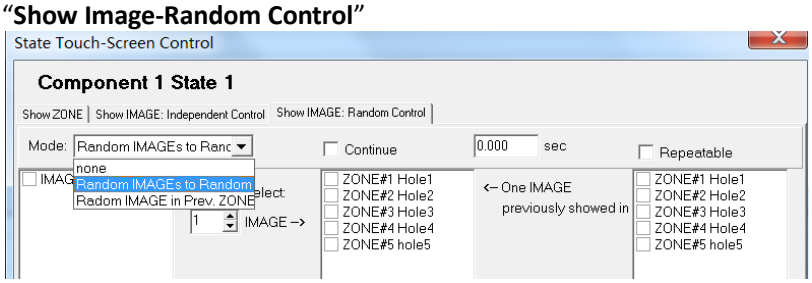


“Show Image-Independent control”





“Show Image-Random Control”

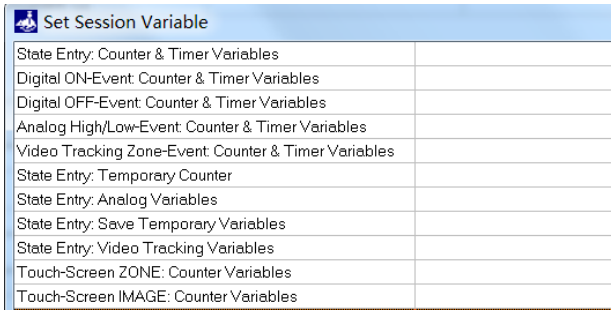


5.2 Set Session Variables

Double click the line of the ”Session Variables” in the state panel.

Upon entering a state, the following session variables can be operated:

- State entry: Counter& Timer Variables
- State entry: Temporary Counter
- State entry: Analog Variables
- State entry: Save Temporary Variables
- State entry: Video Tracking Variables


Set Session Variable
State Entry: Counter & Timer Variables
Digital ON-Event: Counter & Timer Variables
Digital OFF-Event: Counter & Timer Variables
Analog High/Low-Event: Counter & Timer Variables
Video Tracking Zone-Event: Counter & Timer Variables
State Entry: Temporary Counter
State Entry: Analog Variables
State Entry: Save Temporary Variables
State Entry: Video Tracking Variables
Touch-Screen ZONE: Counter Variables
Touch-Screen IMAGE: Counter Variables

5.2.1 State Entry: Counter& Timer Variables

When a state is started, the counter and timer variables can be operated.

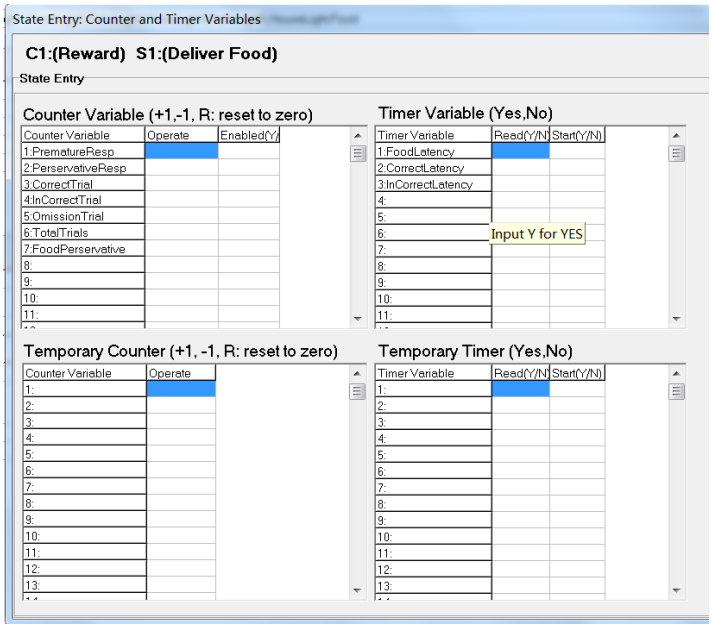
Counter variable has three operations:

- +1: increase one
- -1: decrease one
- R: reset to zero

Counter variable can be “Enabled” or “Disabled”. In the start of a session, all counter ariables are “Enabled“.

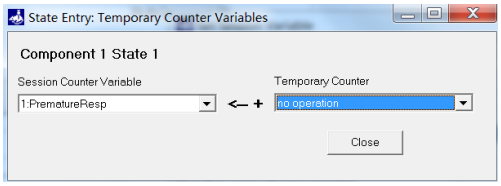
Timer variable has two operations:

- Start the timer
- Read/Stop the timer



5.2.2 State Entry: Temporary Counter

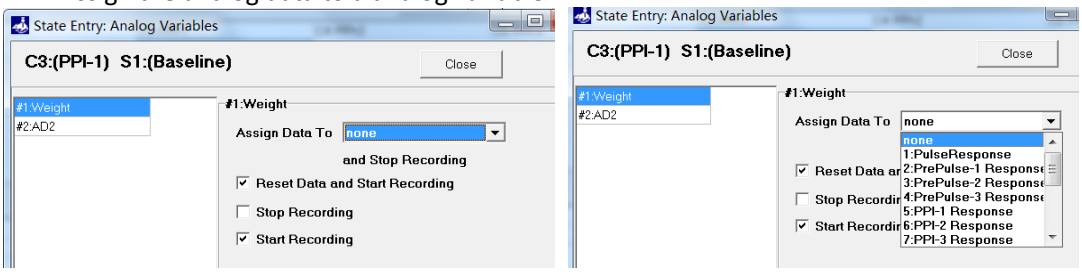
Temporary data are not saved in the data file automatically, but can be assigned to a session counter variable.



5.2.3 State Entry: Analog Variables

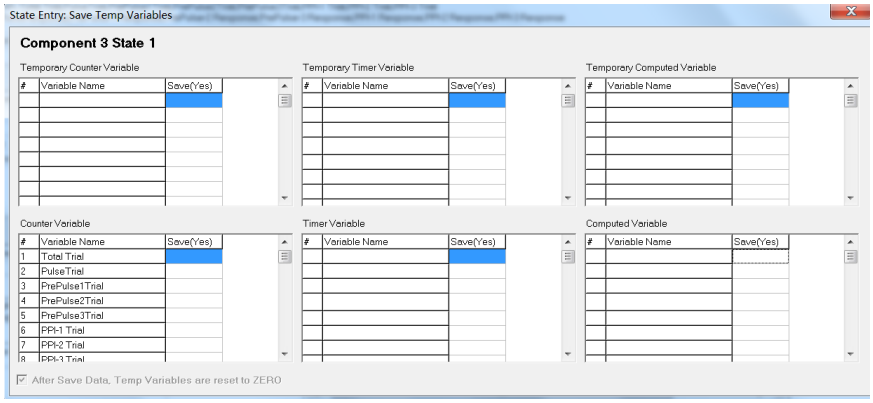
Analog variable has following operations:

- Reset data and start recording
- Start recording
- Stop recording
- Assign the analog data to a analog variable.



5.2.4 State Entry: Save Temporary Variables

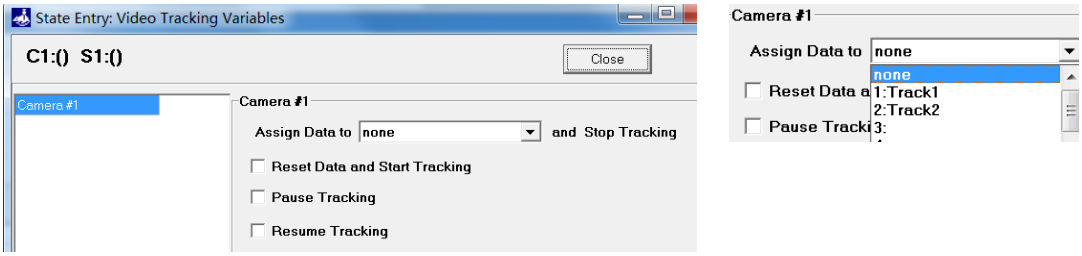
Save temporary variables are used to record block data. Each time the data is saved, the variable is reset to zero.



5.2.5. State Entry: Video Tracking Variables

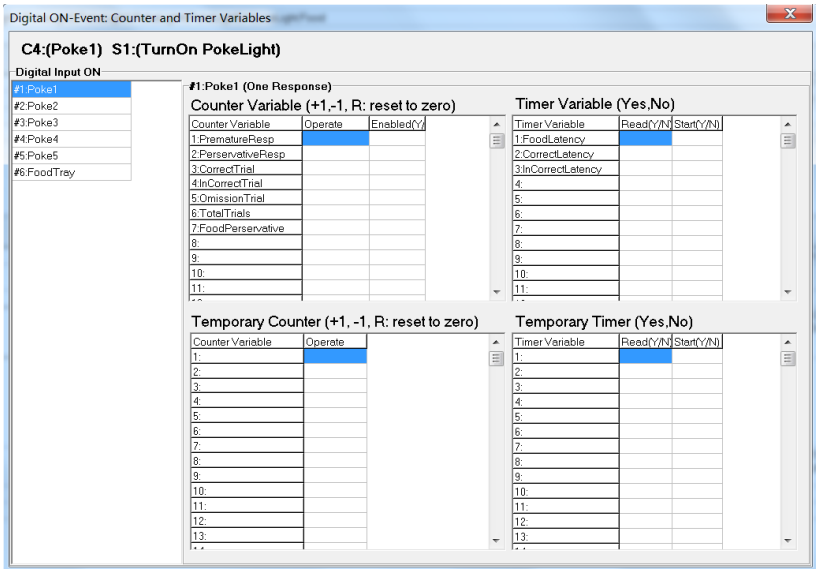
Vide tracking variable has following operations:

- Reset data and start tracking
- Pause tracking
- Resume tracking
- Assign the tracking data to a variable

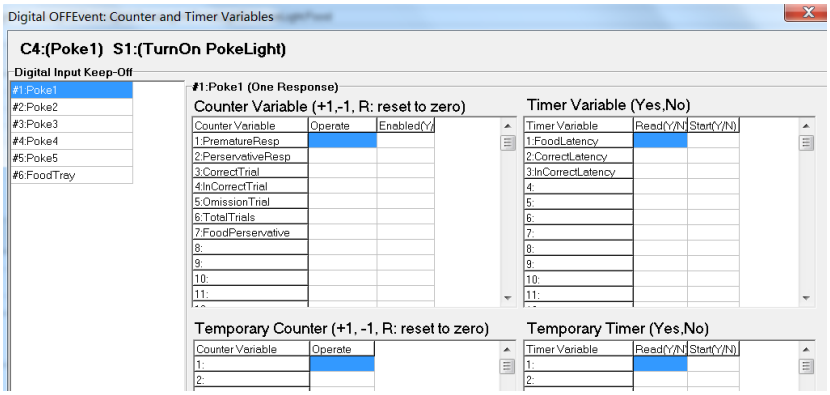


When a state is active, the session variables can also be operated by the following events:

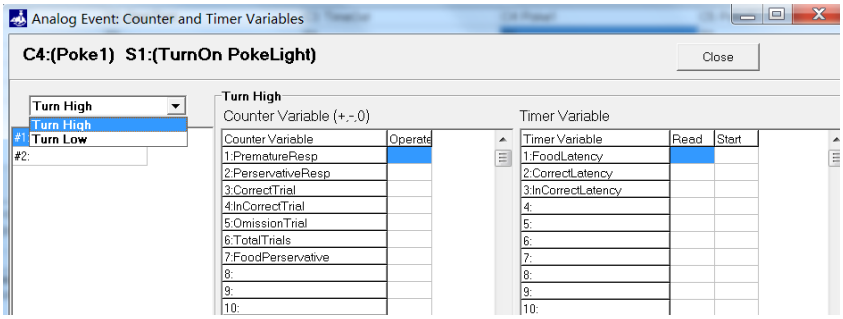
5.2.6. Digital ON-Event: Counter &Timer Variables



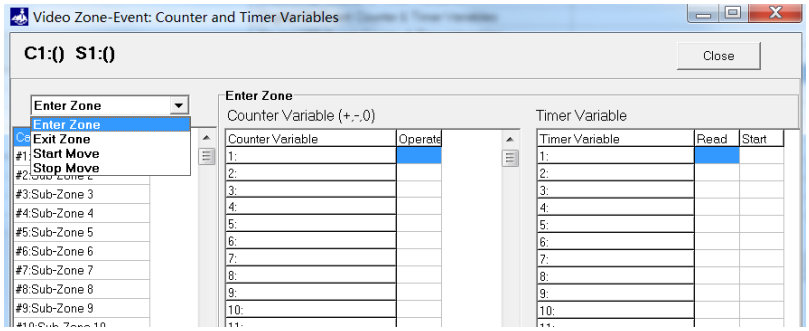
5.2.7. Digital OFF-Event: Counter &Timer Variables



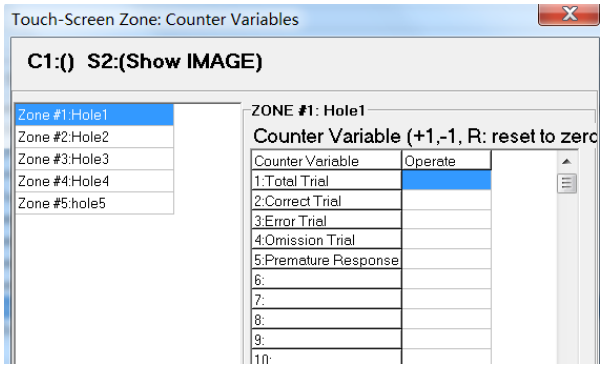
5.2.8. Analog High/Low Event: Counter &Timer Variables



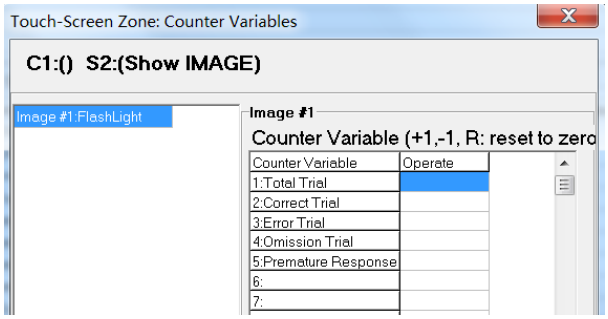
5.2.9. Video Tracking Zone-Event: Counter &Timer Variables



5.2.10. Touch-Screen ZONE: Counter Variables

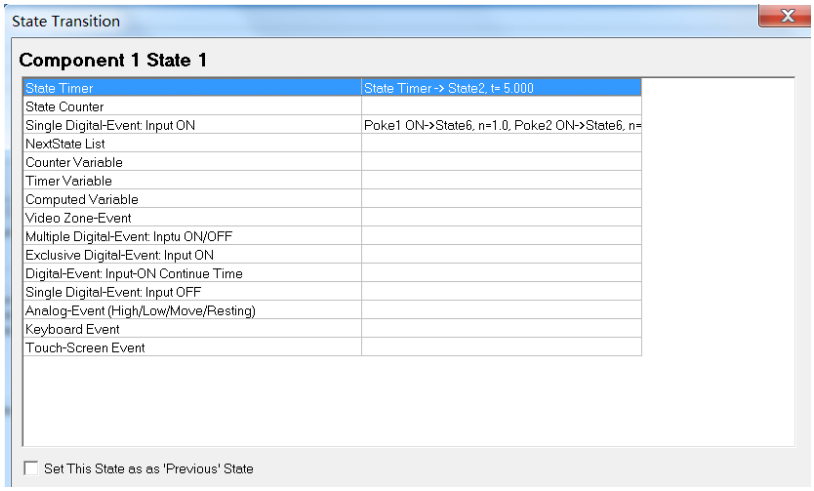


5.2.11. Touch-Screen IMAGE: Counter Variables



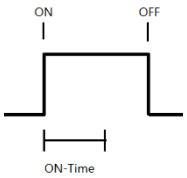
5.3 State Transition

Double click the line of the ”State Transition” in the state panel.



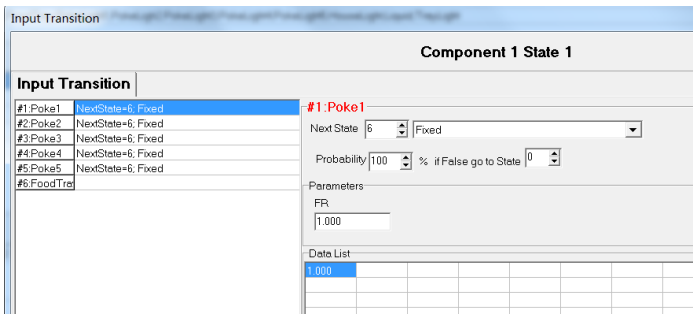
There are 15 events to trigger a state transition.
Single digital (switch) input signal can trigger three state transitions:

- ON: the start of input signal
- Off: the end of input signal
- On-time: the input signal is on for a period of time



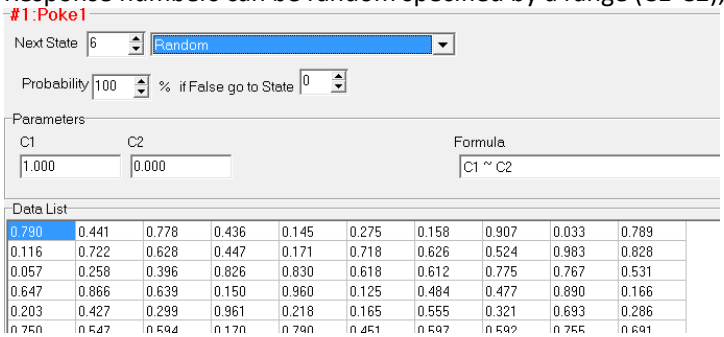
5.3.1 Single digital-event: input-ON

When response (lever, nose poke, etc.) requirements (number of inputs) are met, active state moves to “Next State” with a probability.

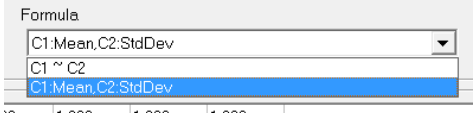


Response numbers to trigger a state transition can be fixed (fixed ratio), in this example, One response in Poke1 goes to state 6.

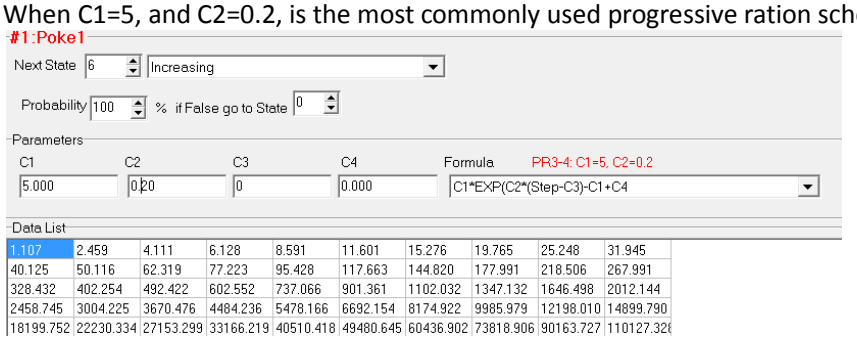
Response numbers can be random specified by a range (C1-C2),



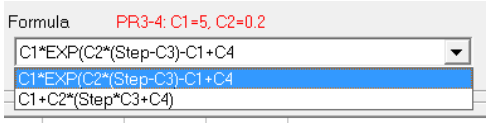
or mean ± standard deviation.



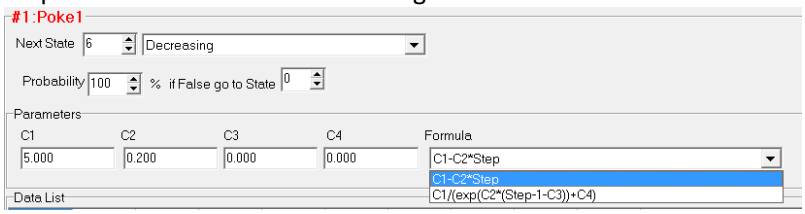
Response numbers can be increasing specified by two formulas:
(1) C1*EXP(C2*[step-C3]-C1+C4)
When C1=5, and C2=0.2, is the most commonly used progressive ration schedule named PR3-4.



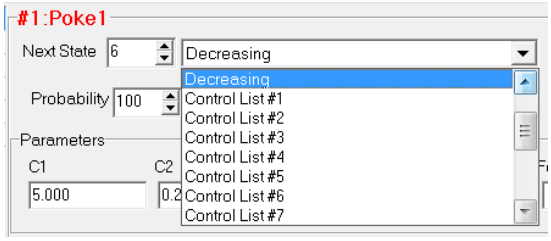
(2) C+C2*(step-C3+C4)



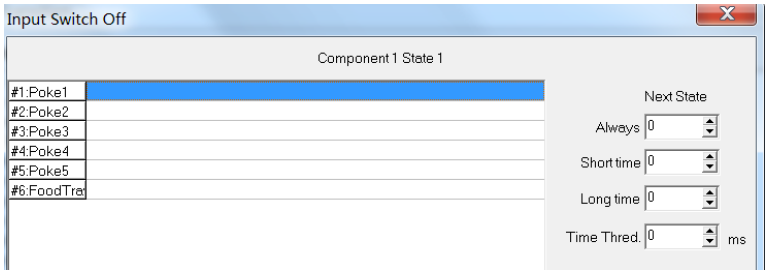
Response numbers can be decreasing with two formulas:



Response numbers can be also be read from a control list.



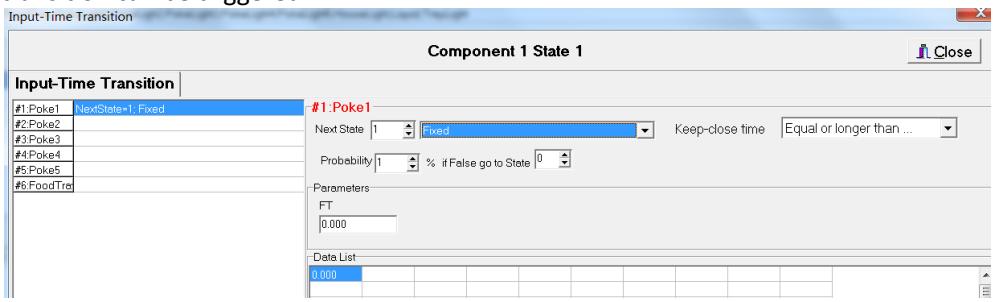
5.3.2 Digital Event: input-off



Digital input signal is OFF triggers a state transition. There are three “Next State”s.
“Always” mean each OFF can triggers state transition.
“Short time” means when input duration is shorter than the “Time Thred”.
“Long time” means when input duration is longer than the “Time Thred”.

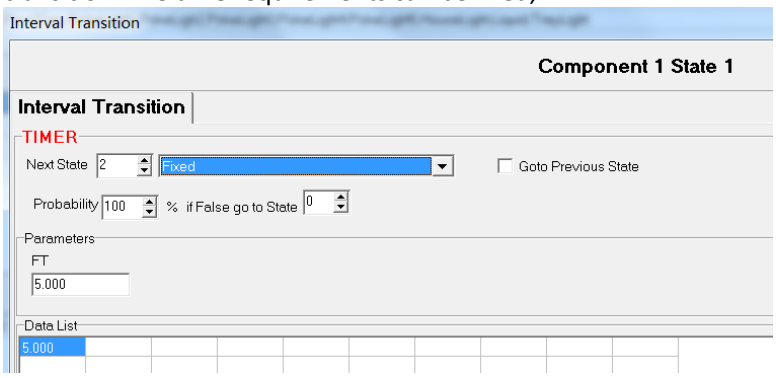
5.3.3 Digital-Event: Input-ON time

When the digital signal is ON, and when the ON duration lasts for a period of time, a state transition can be triggered.

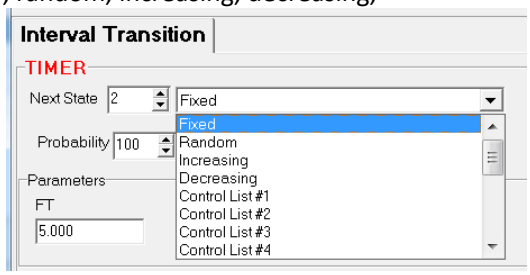


5.3.4 State Timer

Each time a state begin, an internal state timer is turned on, this timer can triggers a state transition. The time requirements can be fixed,

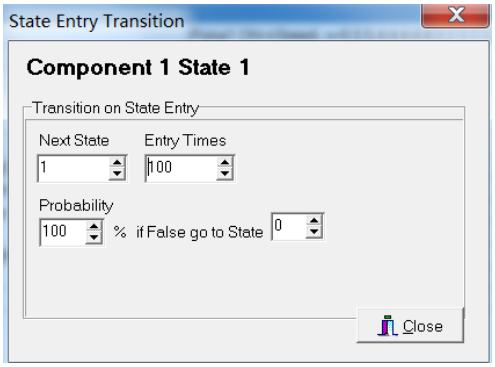


, random, increasing, decreasing,



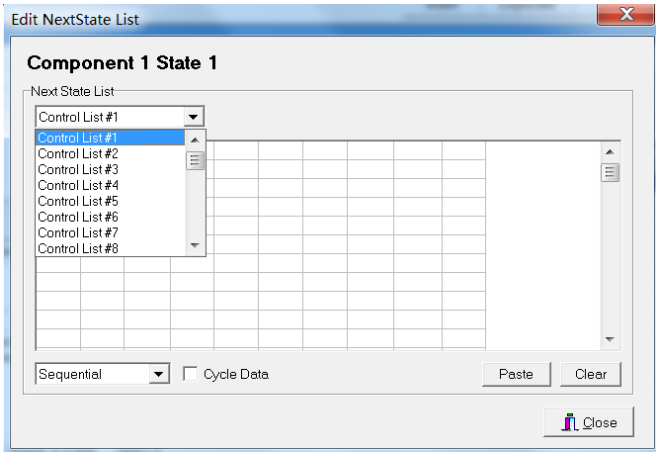
5.3.5 State Counter

Each state also has a counter, each state entry increases the counter. This counter can triggers a state transition with a probability and reset the counter.



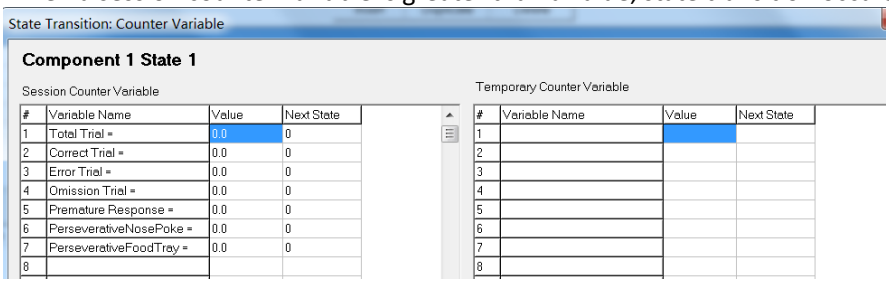
5.3.6 Next State List

Next State List is a state transition without condition. State will automatically goes to a state read from a control list.



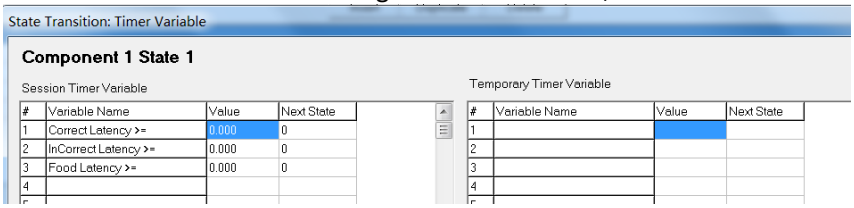
5.3.7 Counter Variable

When a session counter variable is greater than a value, state transition occurs.


State Transition: Counter Variable
Component 1 State 1
Session Counter Variable
Temporary Counter Variable
Variable Name Value Next State
1 Total Trial = 0.0 0
2 Correct Trial = 0.0 0
3 Error Trial = 0.0 0
4 Omission Trial = 0.0 0
5 Premature Response = 0.0 0
6 PerseverativeNosePoke = 0.0 0
7 PerseverativeFoodTray = 0.0 0
8

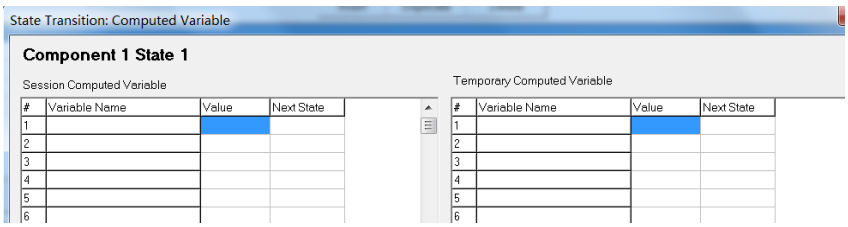
5.3.8 Timer Variable

When a session timer variable is greater than a value, state transition occurs.


State Transition: Timer Variable
Component 1 State 1
Session Timer Variable
Temporary Timer Variable
Variable Name Value Next State
1 Correct Latency >= 0.000 0
2 InCorrect Latency >= 0.000 0
3 Food Latency >= 0.000 0
4

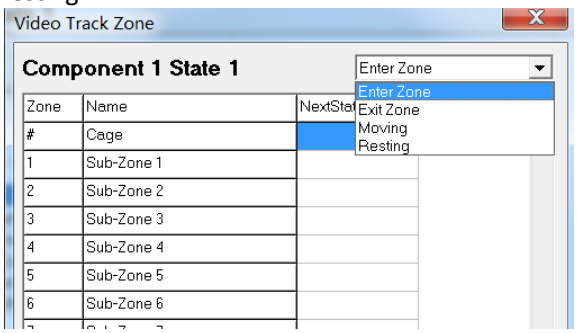
5.3.9 Computed Variable

When a session computer variable is greater than a value, state transition occurs.



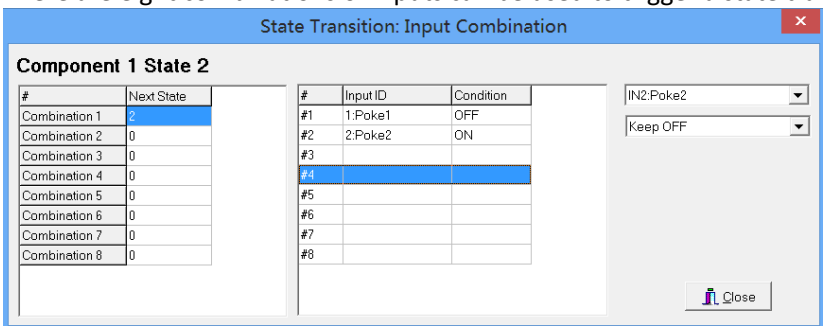
5.3.10 Video Tracking Zone

Video tracking events can trigger state transition. Such as enter a zone, exit a zone, moving or resting.



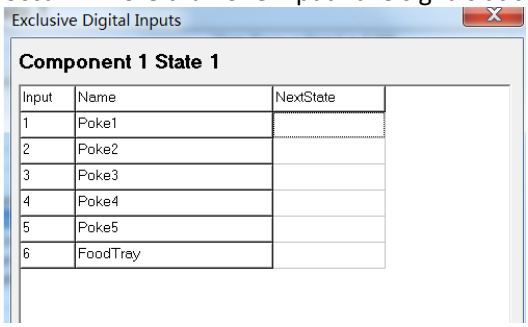
5.3.11 Multiple Digital Event: Input ON/Off

There are eight combinations of inputs can be used to trigger a state transition.



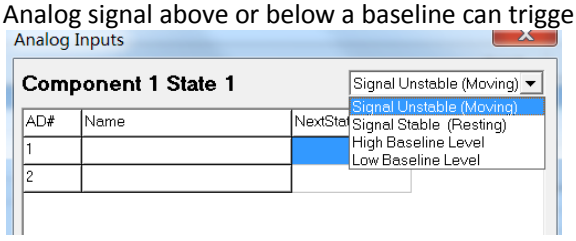
5.3.12 Exclusive Digital Event: Input ON

If the system has many digital inputs, at one, if only one input has signal, a state transition can occur. If more than one input have signals at the same time, state transition is not allowed.



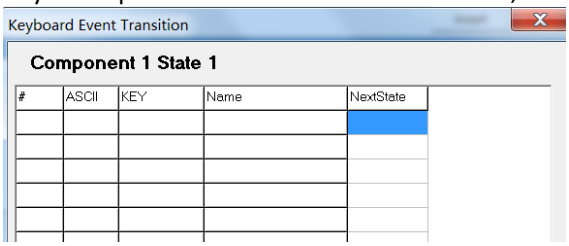
5.3.13 Analog Event: high/low

Analog signal above or below a baseline can trigger state transition.



5.3.14 Keyboard Event

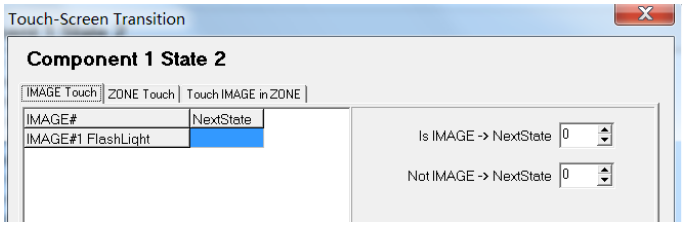
Keyboard press can be used to record events, and these events can trigger state transitions.



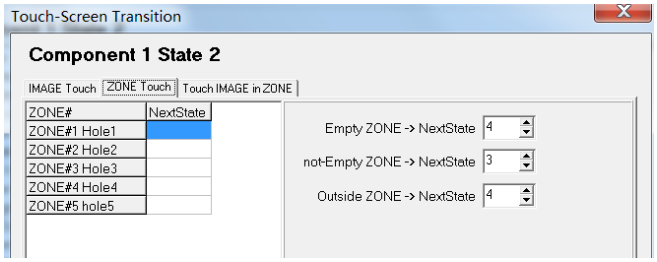
5.3.15 Touchscreen event

In touchscreen system, “ZONE” is a defined region of the screen, and “IMAGE” is a shape, text, or picture that can be showed in a “ZONE” or in a given position (x,y) in the screen.

When the animal touch the screen, a state transition can occur if the touched position has or not has an “IMAGE”.

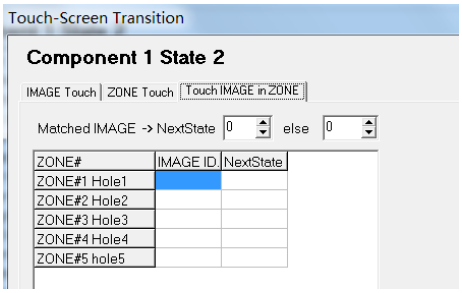


If touched position both inside or outside a “ZONE” can trigger state transition.
Empty “ZONE” means no “IMAGE” has been shown in this “ZONE”;
Not-Empty “ZONE” means an “IMAGE” has been shown in this “ZONE”.



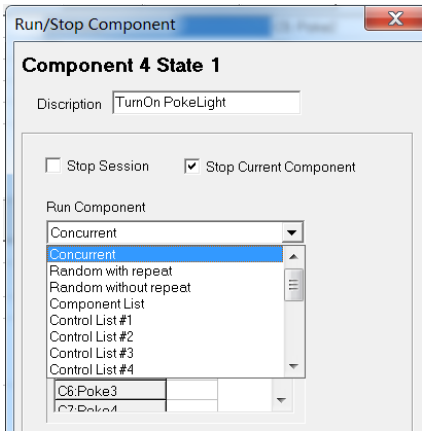
Touch a “ZONE” with a specified “IMAGE” can trigger a state transition.

“ZONE” and “IMAE” can be paired, when the touched “ZONE” is showed with the matched or unmatched “IMAGE“, state transitions occur.



5.4 State Exit

Double click the line ”Exit State” in the state panel.



“Stop session” is used to stop a “run” of the method and save the data to a data file.
“Stop component” is used to stop the current component.

- If the session has only one component, the program will wait until the session duration is time-out.
- If the session has more than one component, you can specify the running status of other components.

6. Sample Methods

(in preparation)

6.1 Second order schedule

6.2 Five-choice serial reaction time task

6.3 GO/NO-GO

6.4 Pre-pulse Inhibition of acoustic startle reflex (PPI)

6.5 Active-avoidance: shuttle box

6.7 Touch-screen: 5-CSRT